Sept 22nd

Clustering - create groupings/assignment of objects s.t
- similar to each other
- dissimilar to other clusters

Uses
- outlier / Anomaly detection (data cleaning | credit card fraud | spam filter)
- filling gaps in Data (same marketing strategy for similar value/ infer probable values for gaps)

Clustering Problem
- SIMILAR DATA POINTS IN SAME CLUSTER
- DISSIMILAR DATA POINTS IN DIFF CLUSTERS

- similar meaning?
- how to find clusters?
- whats a good cluster?

TYPES
- Partitional (each obj → 1 cluster)
- Heirarchical (set of nested clusters organized in a tree → (phylogenetic tree based on DNA/genome seq)
- Density-based (defined based on local density of points)
- Soft clustering (each point assigned to every cluster w/ a probability → (weight of species, → speak more in terms of likelihood)

PARTITIONAL

n data points, set to have k clusters up front

GOAL: partition n into k while maximizing similarity w/in cluster + dissim b/w clusters [intracluster]

intracluster dist : $\underbrace{\sum_{k=1}^{K} \underbrace{\sum_{x_i, x_j \in C_k} d(x_i, x_j)}_{\text{pairwise distance per cluster}}}_{\text{summed over all clusters}}$

NOT efficient so FIND CENTER OF MASS PER CLUSTER [CENTROID]

when d is euclidean, the centroid of m points is the average of the points. (use size of cluster $|C_k|$ to weight the distances to the centroid.)

k-means

$$\sum_i^k \sum_{x \in C_i} \|x - \mu_i\|_2^2$$

FIND k points (means) that minimize the cost function.

Note: easy if k=1 or k=n

Note: if x_i in greater than 2 dim ⇒ NP-hard.

K-means (Lloyd's Algo)
1. Randomly pick k centers $(\mu_1 \cdots \mu_k)$
2. Assign each point in dataset to its closest center
3. Compute the new centers as means of each clusters.
4. Repeat 2,3 until convergence.

Proof by Contradiction
Suppose it does not converge then
1. minimum of cost func is only reached at limit
CONTRADICTION: only finite data points, so we can't have infinite iterations
2. cycle
CONTRADICTION: suggests we increase cost func at some point.

CONCLUSION: ALWAYS converges

Note: will not always converge to optimal
↳ all depends on k points you start w/.

- Farthest First Traversal

choose centers as far from others → BUT does poorly w/ outliers.

- K-means ++
1. start w/ random center
2. let D(x) be dist b/w x and centers selected so far, choose next center w/ prob. proportional to $D(x)^a$

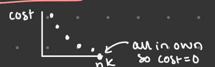
$a=0$ → random $a=2$ → k-means ++ $a=\infty$ → farthest first traversal

generate a rand num 0 to N and set intervals w/ prob of each point, and the value to be picked will be proportional
↳ $\sum_{x \in D} D(x)^2$

K-means limitation
- splits large clusters
- dislikes nonglobular cluster shape
- dislikes varying density

How to choose right k?
1. iterate through different values of k (elbow method)
2. use empirical / domain-specific knowledge.

ELBOW METHOD



ELBOW METHOD MAXIMIZES COST AND NUMBER OF CLUSTERS.

Note: k-means is good for spherical gaussians

K-mean variations

- K-medians (uses L_1 norm / manhattan dist)
- K-medoids (any dist func + centers must be in dataset)
- weighted K-means (each point has a different weight when computing mean) → good for outliers.